Knowledge-Based Systems 242 (2022) 108317

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Real noise image adjustment networks for saliency-aware stylistic color retouch

Bo Jiang [a], Yao Lu [a,*], Guangming Lu [a,*], David Zhang [b,c]

[a] Department of Computer Science and Technology, Harbin Institute of Technology at Shenzhen, Shenzhen 518057, China
[b] School of Science and Engineering, The Chinese University of Hong Kong at Shenzhen, Shenzhen 518172, China
[c] School of Data Science, Chinese University of Hong Kong Shenzhen, China

ARTICLE INFO

Article history:
Received 13 October 2021
Received in revised form 25 January 2022
Accepted 25 January 2022
Available online 18 February 2022

Keywords:
Automatic image adjustment
Real noise image adjustment
Adaptive denoise
Stylistic color retouch
Saliency-aware retouch

ABSTRACT

Automatic Image Adjustment (AIA) mainly aims to realize stylistic color retouch in images. Recent years have witnessed unprecedented success in learning-based AIA methods, especially convolutional neural networks (CNNs). However, existing AIA methods usually handle images without real noise from ideal scenarios, resulting in poor retouch performance when processing real noise images. Furthermore, these AIA methods lack attentive capability when learning salient areas to perform stylistic color retouch as human artists do. To address these problems, we first remodel the adjustment task for real noise images to remove the real noise. Then, we further propose the Real Noise Image Adjustment Networks (RNIA-Nets) using saliency-aware stylistic color retouch and adaptive denoising methods. Specifically, the saliency-aware stylistic color retouch predicts visual salient areas to learn stylistic color mapping using a proposed multifaceted attention (MFA) module. The adaptive denoising mechanism effectively predicts the denoising kernel for various real noise images. Eventually, to equitably verify the effectiveness of the proposed RNIA-Nets, a new challenging benchmark dataset collected from real noise images is established. Extensive experimental results demonstrate that the proposed method can achieve favorable results on real noise image adjustment, providing a highly effective solution to practical AIA applications. ***The code and datasets will be released at*** https://github.com/JiangBoCS/RNIA-Nets.

1. Introduction

With the prevalence of digital imaging devices and social networking, image retouching has gradually become an important issue in image adjustment. Traditionally, high-quality adjustment is usually hand-crafted by an experienced artist through extensive labor. Professional retouchers perform repeated retouching operations according to an image's color distribution to obtain satisfactory results. This retouching process, however, is heavily laborious and time-consuming. Hence, research on automatic image adjustment (AIA) methods has high practical application value. AIA aims to emphasize a specific stylistic color of visual art by adjusting the color and tone in images [1]. Due to the highly nonlinear and subjective nature of image color and tone adjustment, it is extremely difficult for AIA methods to discover the appropriate mapping from the source images to the target images [2–6]. This difficulty will directly affect the adjustment performance for retouched images. Consequently, AIA is still an active and challenging research topic in the field of computer vision.

Existing AIA methods can be broadly classified into two categories, *i.e.*, exemplar-based methods [7,8], and learning-based methods [1,9,10]. Exemplar-based methods retouch images by transforming the color distribution of exemplar images into the target images. This process results in the limitation that the performance by this kind of method is highly dependent on the sample images [11], thus leading to inflexibility and unsatisfactory results. Hence, this paper will mainly study learning-based methods.

For the learning-based methods, the mapping function is directly learned from the training data pairs, *i.e.*, from the source images to the target images. Specifically, deep neural networks have been widely employed in the learning-based methods. In the classical frameworks [1,14–18], the image adjustment task is modeled by the following mapping formulation:

$$\hat{I}_{output} = f\left(I_{input}; \theta_f\right), \tag{1}$$

where f represents the mapping function from an input image I_{input} to a mapped output image $\hat{I}_{output}$, and θ_f denotes the

* Corresponding authors.
E-mail addresses: jiangbo_PhD@outlook.com (B. Jiang), luyao2021@hit.edu.cn (Y. Lu), luguangm@hit.edu.cn (G. Lu), davidzhang@cuhk.edu.cn (D. Zhang).

https://doi.org/10.1016/j.knosys.2022.108317
0950-7051/

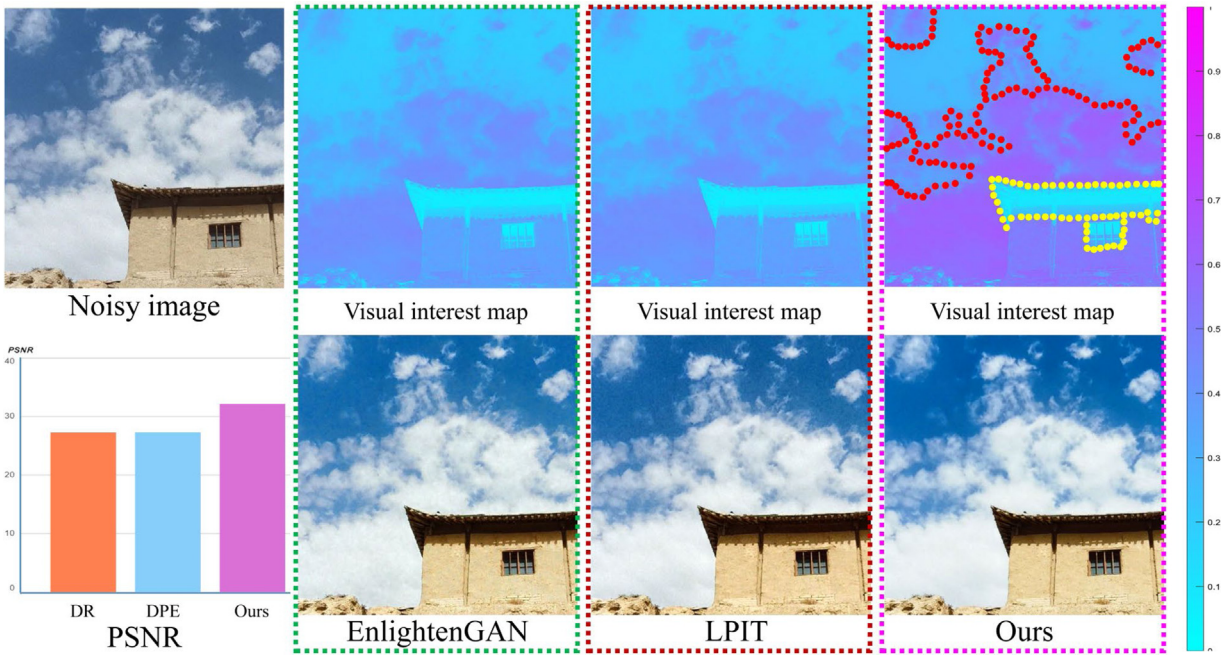


Fig. 1. Examples of visually salient areas in automatic image color adjustment. Our method is more sensitive to the visually salient areas in the input images. The EnlightenGAN [12] and LPIT [13] methods have a weak perception of the visually salient areas.

parameters of the mapping function f. Due to the simplicity and effectiveness of these learning-based solutions, the image-to-image mapping model has become the benchmark framework for evaluating AIA methods [19,20]. In particular, powerful deep CNNs have greatly facilitated the development and improvement of AIA methods. Traditional learning-based AIA methods, however, suffer from two critical problems.

Initially, the performance of AIA is always subjective and it is difficult to perform precise qualitative evaluations. As a human artist retouches the images, visually salient areas in the images can guide the artist to achieve professional image adjustment with high quality. Traditional learning-based AIA methods, however, lack the guidance of this information by the visually salient areas during the stylistic color retouching process. This lack leads to unsatisfactory image adjustment, especially for real noise images from actual scenarios. As shown in Fig. 1, the retouched images produced by traditional learning-based AIA methods, *i.e.*, EnlightenGAN [12] and LPIT [13], are not sensitive to the visually salient areas in the source images. Consequently, it is critical to propose a saliency-aware mechanism that senses the visually salient areas in source images since it will contribute to more professional and high-quality image adjustment.

Furthermore, because of the limitations within the hardware device (especially for mobile devices), the quality of the images captured from such devices will be lost to some extent. These limitations manifest in two respects. First, most of a single image sensor is covered with an array of color filters that capture images, causing unnatural colorization. Second, the image is obtained directly from the image sensor; hence, the resulting image inevitably has complex real scene noise. Therefore, the quality of the retouched images produced by traditional learning-based AIA methods is not satisfactory. Due to this problem, using the traditional assumption (Eq. (1)) to adjust images from real situations may result in poor performance. Because adjusting real noise images is a mixture problem containing denoising and retouching tasks, it is very challenging for traditional AIA methods. For example, denoising destroys the detail/texture information within the images, leading to difficulty during the subsequent retouching step. AIA magnifies the artifacts within the denoised images. Such a mixture problem is important but unfortunately has received insufficient attention. Accordingly, it is vital to propose an adaptive denoising mechanism for the image adjustment network that solves this mixing problem.

Specifically, for the first problem, as demonstrated in the literature [21], the attention mechanism can cause learning-based methods to be more targeted when observing targets. Therefore, inspired by this property of the attention mechanism, we propose a multifaceted attention (MFA) module that automatically predicts visually salient areas in real noise images. The MFA module includes four submodules, *i.e.*, height recursive attention (HRA), width recursive attention (WRA), channel recursive attention (CRA) and weight fusion attention (WFA). HRA, WRA and CRA can recursively retrieve attentive weights from all the extents of input features, leading to the comprehensive perception of visually salient areas. According to the obtained weights from HRA, WRA, and CRA, WFA fuses these weights to produce the final multifaceted attention information. Furthermore, a feedback mechanism is introduced into the MFA block to establish memory during a number of specific training iterations, thus incrementally improving the predictions of visually salient areas in real noise images.

For the second problem, since the traditional assumption cannot handle images from real-world scenarios, we remodel the task of real noise image adjustment. A new assumption is established both for real noise and stylistic color mapping, and this assumption contributes to simultaneous denoising and color retouching for the adjustment of real noise images. To adaptively denoise the real noise images inside the image adjustment network, we review and analyze the conventional methods used for image restoration that are based on deep learning frameworks [22–25]. Inspired by these works, we safely deem that predicting a denoising kernel can effectively restrain the real noise. In addition, the feature retrieved from the input images can also be utilized to predict the denoising kernel, leading to a pertinent denoising process according to various input images. Finally, by applying these two strategies, an adaptive denoising mechanism inside the image adjustment network is proposed. This mechanism can dynamically predict the denoising kernel

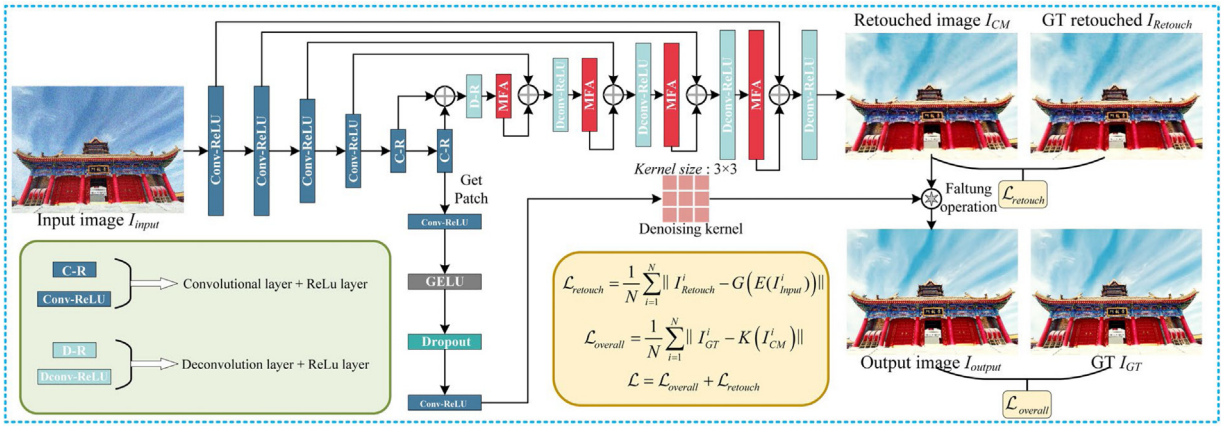


Fig. 2. The overall framework of RNIA-Nets. RNIA-Nets has two branch structures. One branch is a decoder with a saliency-ware stylistic color retouching mechanism. It mainly uses the multifaceted attention (MFA) block with a feedback mechanism to aggregate pyramidal features and predict visually salient areas to gradually generate the adjusted noisy images. The other branch is a structure with an adaptive denoising mechanism, which mainly suppresses real noise in the adjusted noisy images using the adaptive predictive denoising kernel. *C-R* denotes *Conv-ReLU*, and *D-R* represents *Dconv-ReLU*. $E(\cdot)$ is the encoder that extracts features, $G(\cdot)$ represents the decoder with the MFA block branch model, and $K(\cdot)$ is the denoising operator.

to achieve pertinent adjustment for different real noise images. Accordingly, by combining the remodeled assumption and the proposed mechanism illustrated above, this paper finally introduces the Real Noise Image Adjustment Networks (RNIA-Nets), whose framework is depicted in Fig. 2.

The main contributions of this work are summarized as follows:

- We remodel the task of real noise image adjustment based on real noise and stylistic color projection. Based on this assumption, we propose the Real Noise Image Adjustment Networks (RNIA-Nets), which jointly introduces an adaptive denoising mechanism and saliency-aware stylistic color retouching for real noise images. To the best of our knowledge, this is the first attempt to adjust real noise images by concurrently denoising and retouching stylistic colors in a single framework.
- We propose a saliency-aware stylistic color retouching mechanism that can comprehensively and incrementally sense the visually salient areas from all the extents of retrieved features using a proposed MFA module with feedback learning. This mechanism can contribute to more professional and high-quality retouching for real noise images.
- We establish a new challenging benchmark dataset as common practices. Meanwhile, extensive experimental results demonstrate that the proposed method can produce favorable results in terms of real noise image adjustment.

The rest of this article is organized as follows: Section 2 briefly reviews the color adjustment methods used for images as well as the denoising method based on learning. The proposed method is described in Section 3, which includes the remodeled assumption for real noise image adjustment and the entire proposed framework of RNIA-Nets with both the relevant saliency-aware stylistic color retouching mechanism and adaptive denoising mechanism. Section 4 discusses the experimental results of the performance comparisons, which use various methods to process five specific color style datasets, and the ablation experiments using the proposed method. Finally, Section 5 presents the conclusions and possible future research.

2. Related work

This work is related to image color adjustment and image denoising methods. In this section, we mainly review classical and widely used methods.

2.1. Image color adjustment

There are two types of methods for solving the problem of automatic image color adjustment: (i) exemplar-based image color adjustment methods and (ii) learning-based image color adjustment methods.

2.1.1. Exemplar-based image color adjustment methods

In this kind of method, such as those in [7,8,11], the global color distribution of a source image is warped to mimic an exemplar color style. Then, the color distribution of the exemplar images is transferred to that of the target images. Recent research has implemented (semi)automatic selection of exemplar images using image retrieval, and this process has contributed to improving the matching degree [7,8,11] between the source and target images. Ref. [11] investigated the effect of exemplar images on color adjustment. Through extensive experiments in this study, it was proven that exemplar-based image color adjustment methods can provide expressive adjustment and diverse stylizations on color retouching. This kind of color adjustment method, however, is heavily dependent on the exemplar images and the matching degree between the source images and the target images, which limit the improvement in terms of the image color adjustment performance.

2.1.2. Learning-based image color adjustment methods

Different from previous methods, the learning mechanism is imported into learning-based image color adjustment methods to automatically predict the pixel color of specific styles using the given features of color and semantic context. Since these learning-based color adjustment methods are data-driven learning methods, Ref. [26] established an input/retouched image pairs dataset, named MIT-Adobe FiveK, and trained the color adjustment model using a supervised paradigm. Due to the powerful learning capability of deep neural networks, the frameworks of these color adjustment methods extensively employ them to improve the performance of color adjustment. For instance, [17,27] utilized deep CNNs to learn and produce specific stylistic color adjustments in the target images. Yan et al. [10] proposed a deep framework that uses local descriptors to successfully capture complex image styles. This method includes two stages. In the first stage, the source image is segmented using a certain number of superpixels and then encoded using local descriptors. In the second stage, the obtained coding information is input into the

deep neural networks to learn the mapping from the source images to the target images. Since the number of superpixels determines the perception of the visually salient areas, this parameter will hinder the performance improvement of image color adjustment.

2.2. Image denoising method

In recent years, with the development of deep learning, denoising performance based on learning [28–30] has shown great promise. There are some works that can denoise images containing real noise, including MPRNet [22] and MIRNet [23]. Specifically, CBDNet [30] first obtains pseudoraw images from camera pipelines. Then, the synthesized near-real noise images are generated by supplementing the pseudoraw images with heteroscedastic Gaussian noise. This denoising method can simulate more than 200 various camera response functions to generate noisy images with different characteristics. Moreover, CBDNet could be alternately trained with both real and synthetic noise images to overcome overfitting in noisy models to some extent. Therefore, this alternate training scheme, we argue, may cause instability during the training process due to the different distributions of the training data. Furthermore, since the real noise is coupled in the image content, the real noise distribution may change as the image content changes. This change will result in less pertinence from various image contents during the denoising process and thus restrict the restoration performance for noisy images. Recently, during the task of burst sequence image sequence denoising, multiple denoising kernels have been predicted using the time information provided by the burst sequence [31]. Compared to this work, our denoising scheme uses a dedicated denoiser to retrieve pertinent semantic features that are used to predict the adaptive denoising kernels instead of relying on the temporal information provided by the burst image sequences. Then, the Iterative Kernel Correction (IKC) scheme is proposed to iteratively refine the Super-Resolution (SR) kernel estimation and High Resolution (HR) recovery [32]. The IKC is limited to isotropic Gaussian SR kernels with different variances. However, because our denoising kernels are adaptively predicted using the retrieved features containing complex real noise information from real noise images, the predicted denoising kernels are general and not limited to isotropic Gaussian kernels with different variances.

2.3. Attention mechanism

The attention mechanism [33–36] has been proven to improve the performance of various computer vision tasks such as person re-identification [37,38] and image recovery [39,40]. A successful example is ECA-Net [34], which simply compresses each 2D feature map to effectively establish interdependence between channels. CBAM [33] further attempts to exploit positional information by reducing the channel extent to compute spatial attention. CC-Net [36] exploits a nonlocal mechanism to capture different types of spatial information. Axial [35] sparsifies the attention matrix using different sparsity patterns. *Different from these approaches, our approach can distinguish different features and pixel areas, which may provide extra flexibility when processing different types of information and expand the CNNs' representation ability.*

For real noise image adjustment, we propose Real Noise Image Adjustment Networks (RNIA-Nets). Different from the AIA method [12,13], to maximize the capture of the visually significant areas in the images, we propose a salient-aware stylistic color retouching mechanism. At the same time, we use the adaptive denoising mechanism to remove the real noise in the image. This model can provide more professional and high-quality retouching of images.

3. Method overview

For the real noise image adjustment problem, we remodeled the task of real noise image adjustment based on real noise and stylistic color mapping. Based on this assumption, we propose RNIA-Nets to jointly retouch color and denoise real noise images. There are five subsections in this section that introduce the process of establishing new assumptions: the pipeline built based on the new assumption, the salient-aware stylistic color retouching mechanism, the adaptive denoising mechanism, and the loss function.

3.1. New assumption establishment

Before solving the mixture AIA problem, this paper establishes a novel mapping model for real noise image adjustment. The formulation is shown by the following equation:

$$\hat{I}_{output} = f\left(I_{input} \odot n; \theta_f\right) \circledast k, \tag{2}$$

where $\odot$ represents the coupling operation of I_{input} and the real noise n, k represents the denoising kernel, and f is the mapping function from the source images to the stylistic color images. $\circledast$ represents the faltung operation. Mathematically, the corresponding specific style color image $\hat{I}_{output}$ from the source image I_{input} containing real noise can be estimated by solving the following maximum a posteriori (MAP) [41] problem:

$$\min_{I_{GT}} \left\| \left(f\left(I_{input} \odot n; \theta_f\right) \circledast k - I_{GT}\right) \right\|^2 + \lambda \phi\left(I_{GT}\right), \tag{3}$$

where $\left\| \left(f\left(I_{input} \odot n; \theta_f\right) \circledast k - I_{GT}\right) \right\|^2$ is the data fidelity term, $\phi\left(I_{GT}\right)$ is the regularization term (or prior term), and λ is the regular coefficient. Simply speaking, Eq. (3) represents a very important point, *i.e.*, the estimated solution not only adjusts the image but also denoises the image. Therefore, the MAP solution for a mixture AIA with real noise can be formulated as:

$$y = H\left(I_{input}, k, n, \lambda; \theta\right), \tag{4}$$

where H is the function of the MAP inference, and θ denotes the inference parameters of H. We treat the CNN as a discriminative learning solution to Eq. (4) and obtain the following insights based on this assignment.

Since the data fidelity term corresponds to the mapping and denoising processes, accurate joint modeling of mapping and denoising is critical to the success of mixture AIA. However, most of the existing learning-based color adjustment methods actually aim to solve the following problems:

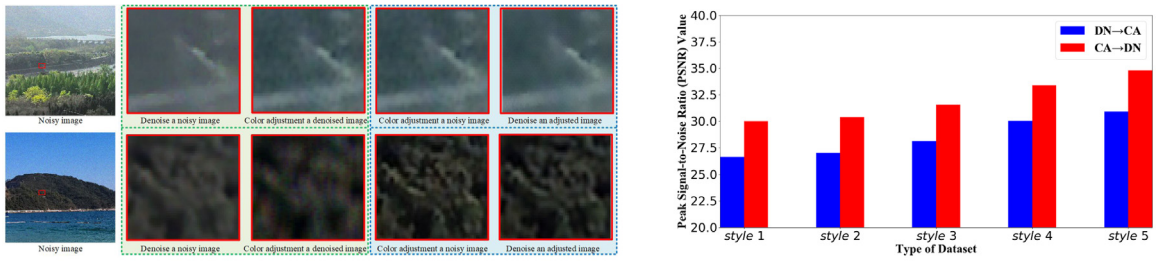
$$\min_{I_{GT}} \left\| \left(f\left(I_{input}; \theta_f\right) - I_{GT}\right) \right\|^2 + \lambda \phi\left(I_{GT}\right), \tag{5}$$

Inevitably, their practicability is very limited. To design a more practical mixture AIA model, it is preferable to learn a mapping function such as Eq. (4), which covers more extensive real-world conditions. Furthermore, to design an end-to-end model, the model must utilize as many external parameters as possible. Therefore, both λ and k can be absorbed by θ_{MAP}. It should be stressed that since existing real noise n can be absorbed into I_{input}, Eq. (5) can be reformulated as:

$$y = H\left(I_{input}; \theta_{AIA}\right), \tag{6}$$

where θ_{AIA} includes the parameters k and λ.

From the perspective of the MAP framework, one can see that the goal of AIA is to learn a mapping function Eq. (6). However, it is not an easy task to directly model Eq. (6) using CNNs. The first reason is that it is difficult to use reasonable visually salient areas for specific color style mapping. The second reason is due to the design of the denoising kernel k within the network. Considering



(a) (b)

Fig. 3. The interactions between color adjustment and denoising. (a) Visual effect of the interaction between color adjustment and denoising. The first row shows that the denoising image contains denoising artifacts and causes the color adjustment to make the denoising artifacts more obvious. The second row shows that the color adjustment is performed first and then the image is denoised; this process reduces the interference of denoising artifacts and provides better results; (b) The influence of *Color Adjustment* → *Denoising* (CA → DN) and *Denoising* → *Color Adjustment* (DN → CA) on the performance after reconstruction. For the dataset used during the test, see Section 4.1 for details.

these two reasons, we propose two mechanisms that explicitly perceive information from real noise and the salient areas in real noise images. In the following two subsections, we propose a saliency-ware stylistic color retouching mechanism, which can produce visually salient areas to improve the quality of retouched images. Furthermore, we propose an adaptive denoising mechanism that effectively denoises real noise images according to various sample features.

Inspired by the contracting-and-expanding image mapping scheme of U-Net [42], RNIA-Nets encodes the input images into a pyramidal structure feature in the deep latent space. RNIA-Nets has two branch structures. One branch is a decoder with a saliency-ware stylistic color retouching mechanism. It mainly uses the multifaceted attention (MFA) block with a feedback mechanism to aggregate the pyramidal features accordingly and to predict visually salient areas to gradually generate the adjusted noisy images; the other branch is a structure with an adaptive denoising mechanism that mainly suppresses real noise in the adjusted noisy images using the adaptive predictive denoising kernel. The overall framework of RNIA-Nets is shown in Fig. 2.

3.2. A novel pipeline for the new assumption

We introduce a novel real noise image adjustment pipeline, that is *Color Adjustment* → *Denoising*. For a given real noise image I_{input}, our pipeline acquires the final high-quality retouched image I_{output} from the real noise images using a composite function:

$$I_{output} = K\left(G\left(E\left(I_{input}\right)\right)\right), \tag{7}$$

where $E(\cdot)$ is the encoder for extracting features, $G(\cdot)$ represents decoder with MFA block branch model, and $K(\cdot)$ is denoising operator. In the proposed pipeline, we first perform color adjustment on the noisy images to obtain the corresponding low-quality retouched version. We then adopt an adaptive denoising kernel to remove noise in the low-quality retouched images to produce the high-quality retouched images. Below, we provide an analysis of why the proposed pipeline's order of color adjustment to denoising is effective.

Here, we recommend adjusting the color first. The main reasons are as follows: (1) After the denoising process, artifacts may be introduced into the denoised images, causing difficulty during the color adjustment task. Denoising artifacts are likely to hide color information, which implies that the color adjustments on noisy images may be affected by unwanted artifacts. (2) Artifacts caused by denoising defects can be avoided during our process, *i.e.*, CA → DN. In other words, color adjustment before denoising can reduce denoising artifacts caused by image degradation limitations. The comparisons of visual effects are shown in Fig. 3(a). The numerical performances are shown in Fig. 3(b). These results prove that the image adjustment processing order in our work is effective. Therefore, we apply this CA → DN order to our method.

3.3. Salient-aware stylistic color retouching mechanism

Visually salient areas from the pyramid-shaped features in the deep latent space of input images can guide the decoder to reconstruct more professional retouched images. Therefore, to retrieve the visually salient areas, we introduce a salient-aware stylistic color retouching mechanism into the decoder, *i.e.*, the MFA block with a feedback mechanism. As shown in Fig. 4a, the proposed MFA block can be expanded using n iterations of feedback numbers in which the output of each iteration is the input to the next iteration. To make the hidden state in the MFA block carry a notion of output, we bind the losses of n iterations together and update the parameters of the MFA block in the back propagation of the networks. The MFA block can distinguish different features and pixel areas, which may provide extra flexibility for processing different types of information and expand the CNN's representation ability. The MFA block contains four parts, including height recursive attention (HRA), width recursive attention (WRA), channel recursive attention (CRA) and weight fusion attention (WFA).

CRA.CRA is mainly used to obtain the attention weights on the channel. We first use global average pooling to produce global spatial information in every channel for channel recursive attention:

$$g_{c1} = H_{p1}(F) = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} F_c(i,j), \tag{8}$$

where $F_c(i,j)$ is the value of the cth channel at position (i,j), H_{p1} is the global pooling function, F is the input feature, and g_{c1} is the output feature. If the size of input feature map is $C \times H \times W$, then, that of output feature map is $C \times 1 \times 1$. As shown in Fig. 4b, in order to obtain the final weights on the channel extent, the squeezed feature pass through two fully connected layers and the attached intermediate ReLU [43] activation functions:

$$Cw_1 = Q_{f1}(g_{c1}) = \delta\left(Fc\left(\delta\left(Fc\left(g_{c1}\right)\right)\right)\right), \tag{9}$$

where Fc is a fully connected layer function and δ is a ReLU function. Finally, we multiply the input F by the weight Cw_1 of the channel element by element:

$$F^*_{c1} = Cw_1 \otimes F. \tag{10}$$

Meanwhile, in order to incrementally learn the channel attention and thus produce more precise visual salient areas, we adopt a recursive method, which uses the same channel attention operation to obtain different orders of channel weights on the basis of the above F^*_{c1} with weight information:

$$\begin{aligned} Cw &= Cw_1 + Cw_2 + Cw_3 \\ &= Q_{f1}\left(H_{p1}(F)\right) + Q_{f2}\left(H_{p2}\left(F^*_{c1}\right)\right) \\ &\quad + \times Q_{f3}\left(H_{p3}\left(Q_{f2}\left(H_{p2}\left(F^*_{c1}\right)\right) \otimes F^*_{c1}\right)\right), \end{aligned} \tag{11}$$

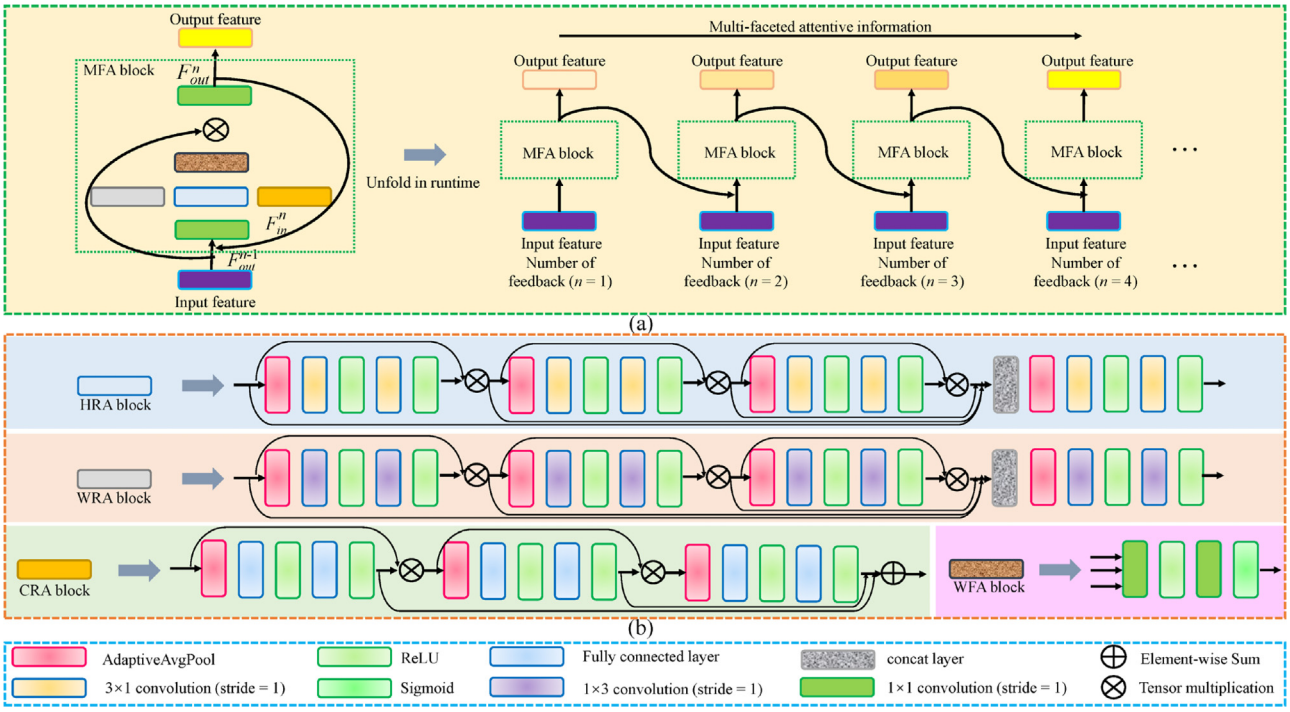


Fig. 4. Multi-Faceted Attention (MFA) Block with a Feedback Mechanism. The proposed MFA block with a feedback mechanism can be expanded using n iterations of feedback numbers, in which the output of each iteration is the input of the next iteration. The MFA block can produce more appropriate visually salient areas, which contain height recursive attention (HRA), width recursive attention (WRA), channel recursive attention (CRA) and weight fusion attention (WFA).

where Cw is the spatial information fusion weights of different orders, Cw_1, Cw_2 and Cw_3 are the weights generated during the channel attention operation, respectively.

HRA and WRA. HRA is mainly used to obtain the spatial feature weights in the height dimension, *i.e.*, the feature weights of HRA is the projection from the feature maps F_{in}^n (in Fig. 4a) produced and squeezed on the width extent. The function of WRA is similar to HRA, primarily to capture the spatial feature weight in the width extent, *i.e.*, the feature weights of WRA is the projection from the feature maps F_{in}^n produced and squeezed on the height dimension. Specifically, we use the average pooling operation to retrieve the spatial information of width and height dimensions for WRA and HRA, respectively. This operation is shown in Fig. 4b.

WFA. Considering that feature weights with different biases can be captured according to different extents, in order and to make the network pay more attention to the visually salient areas, a weight fusion attention is proposed.

The weights (*i.e.*, Cw, Hw, Ww), which are calculated by HRA, WRA and CRA are fed into two convolution layers with ReLU [44] and Sigmoid [45] activation functions in sequence:

$$P_{Cw}, P_{Hw}, P_{Ww} = \sigma\left(M_c\left(\delta\left(M_c(Cw, Hw, Ww)\right)\right)\right), \tag{12}$$

The weights from the activated feature are fused using tensor multiplication:

$$P_{Fw} = P_{Cw} \otimes P_{Hw} \otimes P_{Ww}, \tag{13}$$

where M_c is the convolution layer with convolutional kernel spatial of size 1×1; δ is the Sigmoid activation function; Cw, Hw and Ww have the same size as the corresponding P_{Cw}, P_{Hw} and P_{Ww}, and the sizes of feature maps are $C \times 1 \times 1$, $C \times H \times 1$ and $C \times 1 \times W$, respectively. P_{Fw} is the weight produced by fusing weight information of all features, and the size of the feature map is $C \times H \times W$:

$$F^* = M_c\left(F \otimes P_{FW}\right), \tag{14}$$

3.4. Adaptive denoising mechanism

Since different images may contain different real noise, when these images are input into the network, different representations will be produced. Therefore, the feature extracted from RNIA-Net's encoder is further input into the adaptive predictive denoising kernel structure to generate the pertinent denoising kernels for the specific denoising images. However, the size of the encoder's output feature maps in RNIA-Nets varies with that of input images, as shown in Fig. 5. This variation will cause the predicted denoising kernels to have unstable sizes. Thus, to predict a fixed size for the denoising kernels, we crop the feature maps produced by the encoder into a patch with a spatial size of 16×16 and reuse it as the input to the adaptive denoising structure. To maintain the feature produced by the encoder as much as possible, we randomly crop the feature maps 128 times according to Eq. (15), *i.e.*, 128 patches of the spatial size of 16×16.

$$F_{patch} = M_G\left(F_{in}\right), \tag{15}$$

where $F_{in} \in \mathbb{R}^{C \times H \times W}$ is the input feature of the adaptive denoising branch. C, H and W denote the channels, height and width of the feature maps, respectively. M_G represents the operation of cropping the patch from the feature maps. The procedure of the extracted patches is shown in Algorithm 1. $F_{patch} \in \mathbb{R}^{(128 \times C) \times 16 \times 16}$ denotes the cropped patch feature maps. Specifically, in the process of adaptive prediction denoising kernel, the convolutional layer with the same spatial sizes between the convolution kernels and patch feature maps is used to obtain the global information of the patch feature maps, as shown in Eq. (16).

$$F_g = M_s\left(F_{patch}\right), \tag{16}$$

where M_s is the convolution layer with spatial kernel size 16×16, $F_g \in \mathbb{R}^{16 \times 3 \times 3}$ represents the global feature maps retrieved from the patch feature maps. Next, the Gaussian error linear units [46]

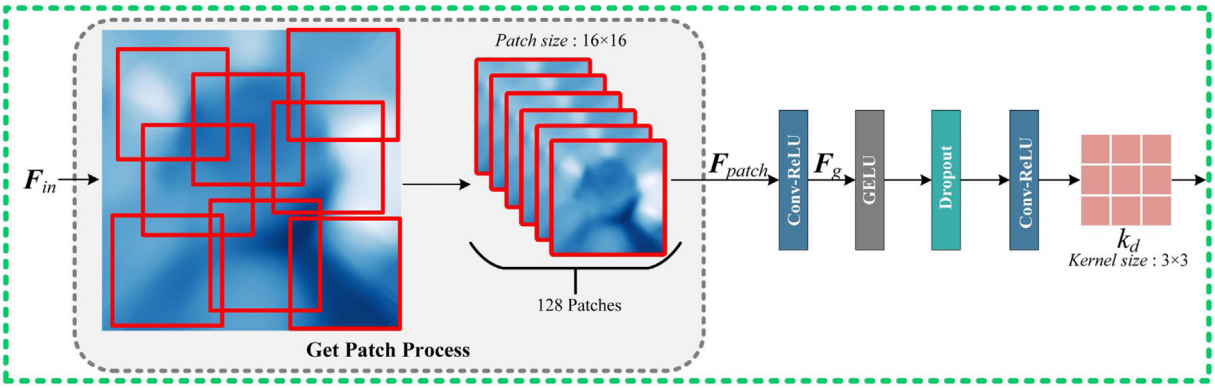


Fig. 5. Adaptive Denoising Mechanism. The main structure of adaptive denoising is composed of: a random patch cropping operator, shown in Algorithm 1, two convolutional layers, an activation layer, a Gaussian error linear units and a dropout layer.

Algorithm 1: Crop Patches From Feature Maps

input : Feature maps F of size $b \times c \times h \times w$;
Number of patches n;
Patch size s .
output: Feature map patches F_{out}.

```
while i <= n do
    i = i + 1;
    id_x = random.randrange(0, w − s + 1);
    id_y = random.randrange(0, h − s + 1);
    F_out.append(F[:, :, id_y : id_y + ps, id_x : id_x + ps]);
end
```

layer and the Dropout layer [47] are used to increase the nonlinearity of the branch and reduce the intermediate feature to decrease redundancy. Finally, the denoising kernel is predicted through the convolutional layer with a spatial size 1×1 of convolutional kernel, as shown in Eq. (17).

$$k_d = M_R\left(\varphi\left(\sigma_g\left(F_g\right)\right)\right), \tag{17}$$

where σ_g and φ are Gaussian error linear units layer and Dropout layer, respectively. M_R denotes the convolution operation of the predictive denoising kernel. $k_d \in \mathbb{R}^{3\times3\times3}$ is the predicted adaptive denoising kernel. The obtained denoising kernel and the images reconstructed by the decoder are performed faltung operation according to Eq. (18) to produce the denoised adjusted images.

$$I_{output} = K\left(I_{CM}\right) = k_d \circledast I_{CM}, \tag{18}$$

where I_{CM} denotes the low-quality retouched images.

3.5. Loss function

In this paper, we propose an end-to-end RNIA-Net and solve the mixture problem with the order of *Color Adjustment* → *Denoising*. Compared with the previous models for handling a single task (e.g., denoising or color adjustment), each part of the proposed network handles a specific task. Our network conducts the color adjustment at the first stage and then denoising at the final stage. We achieve this RNIA-Net by providing the retouched branch with middle-stage supervision in the training process. We calculate the l_1-norm loss on the low-quality retouched I_{CM}:

$$\mathcal{L}_{retouch} = \frac{1}{N}\sum_{i=1}^{N} \| I^i_{Retouch} - G\left(I^i_{Input}\right) \|, \tag{19}$$

where $G(\cdot)$ represents the decoder with MFA block branch model, and I_{Input} represents the input source images. $I_{Retouch}$ is the middle-stage supervision signal, *i.e.*, the low-quality retouched ground truth images generated from the real noisy images (see Section 4.1 for detail), during the training process.

In addition, in order to constrain the denoising branch to be able to adequately denoise the low-quality retouched images I_{CM}, we introduce the overall loss function to ensure the denoising effectiveness, and the loss function is defined as below equation:

$$\mathcal{L}_{overall} = \frac{1}{N}\sum_{i=1}^{N} \| I^i_{GT} - K\left(I^i_{CM}\right) \|, \tag{20}$$

where I_{GT} represents the ground truth noise-free retouched images and $K(\cdot)$ is the denoising operator. The final objective function in our approach is:

$$\mathcal{L} = \mathcal{L}_{overall} + \mathcal{L}_{retouch}. \tag{21}$$

4. Experiments

4.1. Dataset

The images in widely used datasets are mostly sampled from ideal scenarios. This limitation indicates that traditional datasets are not sufficient for evaluating the effectiveness of the proposed adjustment method for differentiating real noise images from actual scenarios. Existing methods for generating noisy images usually directly inject synthetic noise into images. Since the simulated noise is quite different from real noise in actual scenarios, a learning-based AIA method trained on this dataset will have decreased ability to generalize to real noise images from real situations [21].. Hence, constructing an equitable benchmark dataset from real-world scenarios is a challenging issue that needs to be addressed.

For this problem, to fairly evaluate the proposed RNIA-Nets for real noise images, this paper establishes a challenging benchmark dataset, called AIA dataset. We have captured a dataset of 171 noisy images from 30 different scenes using a representative smartphone (Honor 30s, XiaoMi 10 and Huawei P30) cameras. These images are all in the $3,456 \times 4,608$ resolutions. Then, the same sampling method of SIDD dataset [48,49], an open benchmark dataset for denoise competition in CVPR 2020, is employed in the new constructed AIA dataset. A professional photographer used Photoshop to retouch these images and produce the datasets for five different stylistic color effects Fig. 6. Modification operations include local objects/areas with the region selection tool and blending different layers using various modes, *etc.* To reduce the

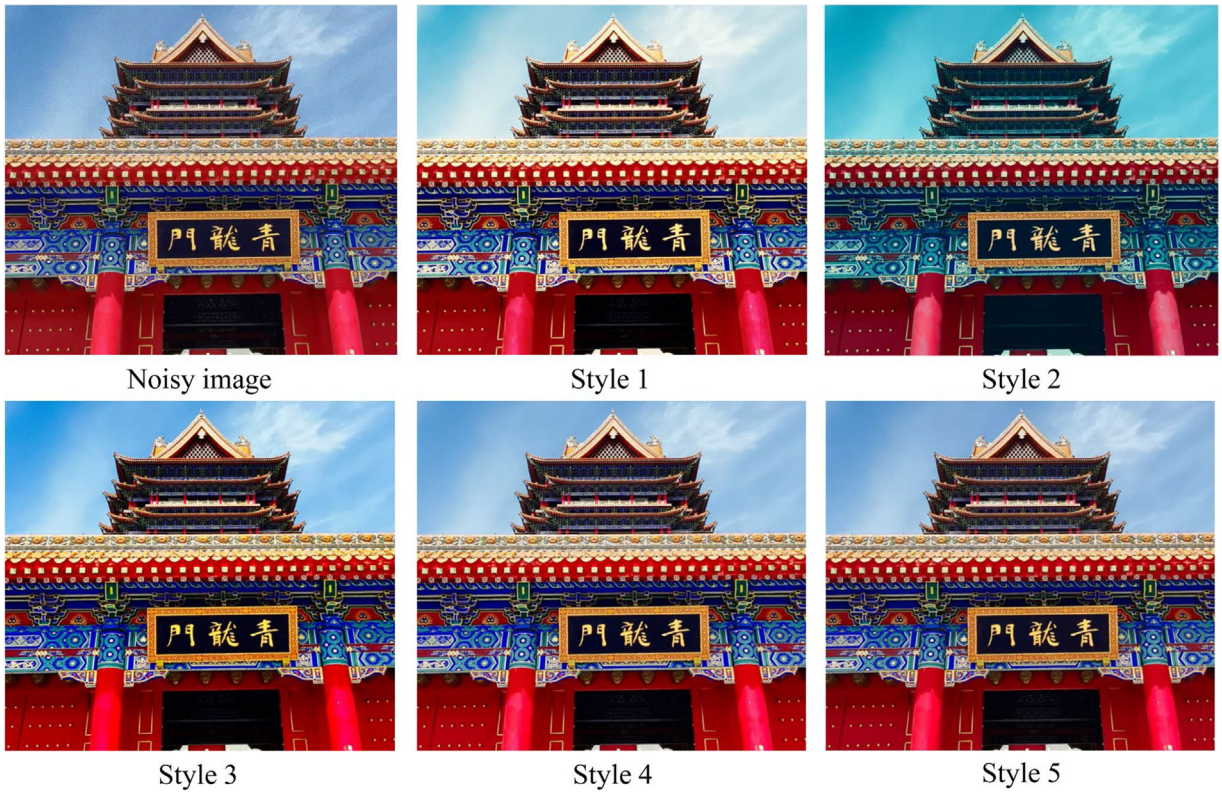


Fig. 6. Examples of five specific style color adjustment effects for the same source image. The professional photographer used Photoshop to retouch these images and produced the datasets for five different stylistic color effects. Retouch operations include changing local objects/areas using the region selection tool and blending different layers using various modes.

subjective difference during the retouching period, the retouching operation process of each specific color style is recorded using the "action" tool in Photoshop, which can process the remaining images in batches. Finally, the dataset is composed of 1,500 training images, 150 verification images and 60 test images with 512 × 512 resolutions from 5 subsets. To provide the intermediate supervision signals for the modification branch during the training process, each subset is divided into three types of images, *i.e.*, noisy images I_{input}, low-quality retouched groundtruth images $I_{Retouch}$, and high-quality retouched groundtruth images I_{GT}. This dataset can greatly satisfy the requirement of impartially evaluating various AIA methods for real noise images. ***The model's code and datasets will be available at*** https://github.com/JiangBoCS/RNIA-Nets.

4.2. Evaluation metric

We employed two commonly-used metrics (*i.e.*, PSNR and SSIM) to quantitatively evaluate the performance of our network in terms of the color and structure similarity between the predicted results and the corresponding expert-retouched images. Although it is not absolutely indicative, in general, high PSNR and SSIM values correspond to reasonable good results.

4.3. Experimental setting

The proposed method is implemented by Pytorch[1] and trained on two Nvidia RTX TITAN GPU with a batch size of 16. Image patches corresponding to the ground truth image and noise image are extracted, respectively. The 512 × 512 size of image pairs is the input of the proposed method. We use Adam [50] optimizer, in which β_1 and β_2 adopt default values of 0.9 and 0.999, respectively. The learning rate is 1×10^{-3}. The network parameters are initialized using the *Kaiming* method in [51].

4.4. Method performance

We compare the proposed method with four methods, including two representative learning-based methods for color adjustment (*i.e.*, LPIT [13] and EnlightenGAN [12]), two classical denoising methods (*i.e.*, MPRNet [22] and MIRNet [23]). For fair comparison, we produce their results using the publicly-available implementations provided by the authors with recommended parameter settings. For the four learning-based methods, we further re-train their models on our dataset to evaluate the best possible results on our test dataset. Our comparisons contain two aspects: visual comparison and quantitative comparison.

Visual Comparison. Firstly, we show a visual comparison of challenging cases in Fig. 7. The input used in these five cases are images with real noise, and each case corresponds to an image with a different specific style of color mapping. All approaches are trained with input noise image/retouched image (ground truth, GT) pairs by five training subsets of specific style colors in our dataset. The following phenomena can be observed from Fig. 7. Since the two representative learning-based color adjustment methods, LPIT [13] and EnlightenGAN [12], do not have the ability to denoise, the noise is obviously amplified in the enhanced images, colored by red and purple boxes. Although the two typical learning-based denoising algorithms, MPRNet [22] and MIRNet [23], can remove part of the noise, the color mapping of specific styles is not accurate, colored by red and purple boxes.

[1] https://pytorch.org.

Table 1
Quantitative comparison between the proposed method and others. The comparison methods include two representative learning-based methods for color adjustment (*i.e.*, LPIT [13] and EnlightenGAN [12]), two classical denoising methods (*i.e.*, MPRNet [22] and MIRNet [23]). The CA → DN reports the results of quantitative analysis of the color adjustment and then the noise removal processing. *Case.I* means LPIT → MPRNet, and *Case.II* denotes LPIT → MIRNet. The DN → CA reports the results of quantitative analysis of the noise removal and then the color adjustment processing. *Case.III* means MIRNet→ LPIT, and *Case.IV* denotes MPRNet→ LPIT.

Type	Metric	Ours	Color Adjustment (CA)			Denoising (DN)			CA → DN		DN → CA	
			DPE	LPIT	EnlightenGAN	FFD	MPRNet	MIRNet	*Case.I*	*Case.II*	*Case.III*	*Case.IV*
Style 1	PSNR	**33.76**	27.43	27.78	28.47	27.26	27.68	30.34	30.01	30.73	28.62	28.51
	SSIM	**0.973**	0.914	0.923	0.943	0.915	0.921	0.963	0.948	0.952	0.947	0.944
Style 2	PSNR	**34.36**	26.30	26.65	29.74	27.38	27.78	30.67	30.39	31.14	29.03	28.89
	SSIM	**0.976**	0.898	0.907	0.963	0.923	0.931	0.969	0.956	0.962	0.957	0.956
Style 3	PSNR	**34.05**	29.07	29.42	24.49	28.74	29.16	31.42	30.56	31.25	29.14	29.06
	SSIM	**0.978**	0.944	0.953	0.957	0.944	0.949	0.971	0.957	0.961	0.958	0.955
Style 4	PSNR	**35.05**	31.69	32.04	32.18	31.27	31.69	31.58	31.42	31.85	31.04	30.92
	SSIM	**0.981**	0.957	0.966	0.969	0.957	0.963	0.968	0.965	0.969	0.959	0.948
Style 5	PSNR	**36.04**	32.76	33.11	33.44	32.63	33.05	33.24	32.31	32.92	31.94	31.83
	SSIM	**0.984**	0.963	0.972	0.971	0.961	0.969	0.967	0.971	0.973	0.965	0.962

We noticed that our method has two key differences compared to other methods. First, our method can restore more details and better contrast on the foreground and background without producing obvious noise spots. Secondly, it can also reproduce a specific style of color, making the adjusted effect look closer to the ground truth.

Quantitative Comparison. To evaluate the learning effectiveness and generalization capability of our network, we quantitatively compare the proposed method with the other methods using the PSNR and SSIM metrics. Tables 1 report the results, where we retrained our network on respective data subsets for each style case as well as others'. The experimental results in Table 1 show that the contrast method (except the proposed method) directly enhances the specific color style of the images with real noise, which will cause the noise to be amplified. We also find from the indicator changes that the learning-based network framework has certain noise suppression capabilities. Meanwhile, from Table 1, we uncover that our method can not only effectively learn to enhance images for a specific style of color, but also can suppress the real noise in the input image. The above results prove the necessity to solve the problem of removing the noise coupled with the corresponding images while enhancing the specific style color of the images. In addition, two processing orders of image adjustment, *i.e.,* CA → DN and DN → CA are evaluated to verify the effect of the orders on the denoising performance. Each processing order is further divided into two categories (*i.e., Case. I*: LPIT → MPRNet, *Case. II*: LPIT → MIRNet. *Case. III*: MIRNet→ LPIT, *Case. IV*: MPRNet→ LPIT). Firstly, compared with *Case. I*, the PSNR of the *Case. II* is 0.64 dB higher than that of the *Case.I* on average. This shows that under the premise of using the same color adjustment method first, the smaller error of the denoising model produces, the better result of image adjustment is. Secondly, compared with *Case. IV*, the PSNR of the *Case. III* is 0.112 dB higher than that of the *Case. IV* on average. This illustrates that using the denoising method first and then using the color adjustment method may amplify the denoising error, resulting in poor adjustment results. Thirdly, compared with DN → CA (*Case. III* and *Case. IV*), the PSNR of the CA → DN (*Case. I* and *Case. II*) is 2.72 dB higher than that of the DN → CA on average. This implies that using the color adjustment method first, and then using the denoising method may avoid amplifying and denoising errors, proving the processing order of CA → DN is reasonable and effective. Finally, compared with CA → DN (*Case. I* and *Case. II*), the PSNR of our RNIA-Nets is 3.714 dB and 3.074 dB higher than *Case. I* and *Case. II* on average. This shows that the structure and strategy of RNIA-Nets we proposed is effective in the task of real noise image adjustment.

Table 2
Two different cascade order denoising and AIA methods, that is, *Color Adjustment* → *Denoising* (CA →DN) and *Denoising* → *Color Adjustment* (DN →CA).

Type	Metric	DN→CA	CA→DN (Ours)
Style 1	PSNR	30.73	33.76
	SSIM	0.952	0.973
Style 2	PSNR	31.14	34.36
	SSIM	0.962	0.976
Style 3	PSNR	32.25	34.05
	SSIM	0.969	0.978
Style 4	PSNR	34.15	35.05
	SSIM	0.973	0.981
Style 5	PSNR	35.02	36.04
	SSIM	0.978	0.984

4.5. Ablation study

Besides the visual results shown in Fig. 7, we quantitatively evaluate the effectiveness of the components in our method. The effectiveness of the components in our method are verified from the following three parts: 1) Effectiveness of the frame structure; 2) Decoder branch with MFA; 3) Effectiveness of the intermediate supervision signals; 4) Denoising kernel branch. ***The settings of the ablation experiment are shown as follows***: The training settings of the ablation study are basically the same as those in Section 4.3, but only the training Style 3 data subset is used in the ablation study, and the effectiveness of each component is evaluated on the validation data subset of Style 3.

Effectiveness of the Frame Structure. The proposed RNIA-Nets architecture is mainly composed of a saliency-aware stylistic color retouch branch and an adaptive prediction denoising kernel branch. The output image of RNIA-Nets is obtained by convolution operation from the results of these two branches, hence, these two branches form a serial structure. Therefore, the RNIA-Nets architecture's order of solving this hybrid problem is to first perform color adjustment and reconstruction and then perform image denoising. At the same time, we evaluate with two different methods of cascading order (*i.e., Color Adjustment* → *Denoising* and *Denoising* → *Color Adjustment*). The results are shown in Table 2. Obviously, the order of *Color Adjustment* → *Denoising* is better than *Denoising* → *Color Adjustment*. From Fig. 8, these frameworks that first use the denoising network and then adjust the cascading order of the color adjustment network will amplify the denoising error and residual artifacts, resulting in poor adjustment effects (red and purple boxes are colored).

Encoder Branch with MFA. In the encoder branch with MFA block, the core component is the MFA block. Since MFA bock is

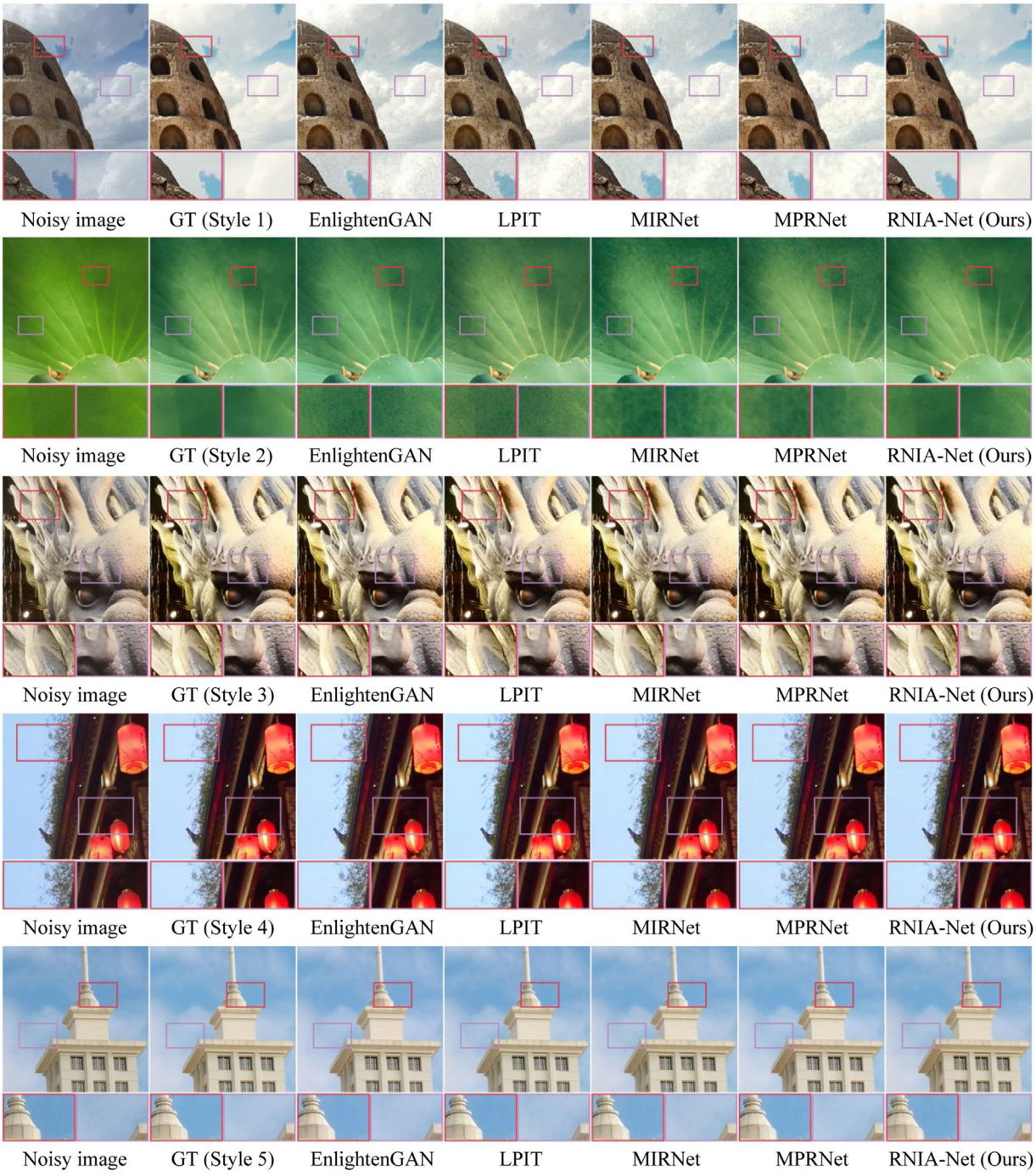


Fig. 7. Visual comparison with four methods on five different style images from our dataset. All approaches are trained with input noisy image retouched image (ground truth, GT) pairs by five training subsets of specific style colors in our dataset.

composed of feedback operation, HRA, WRA, CRA and WFA, we mainly focus on the influence of the above factors on the adjustment effect. In the ablation experiment of the encoder branch with MFA, we retain the denoising kernel branch. In addition, in order to reduce the interference of the intermediate supervision signal during the evaluation of MFA, *we remove the intermediate supervision signal to fairly evaluate the effectiveness of MFA method.*

To clearly and intuitively illustrate the effectiveness of the MFA block with a feedback mechanism, we printed the output feature diagrams of the MFA block. As illustrated in Fig. 9, it is clear that under the same feature maps input, HRA, WRA and CRA adaptively learn completely different weights of feature maps through their own attention dimensions. This different weight information can complement each other to capture salient information from all the extents of sample features. As shown in Fig. 10, in feature maps of the MFA block with the feedback mechanism, visual salient areas are distinguishable and have different weights. These results imply the MFA block with feedback mechanism makes decoder pay more attention to visual salient areas.

From Table 3, we can observe that every factor plays an indispensable role in network's performance. The combination of

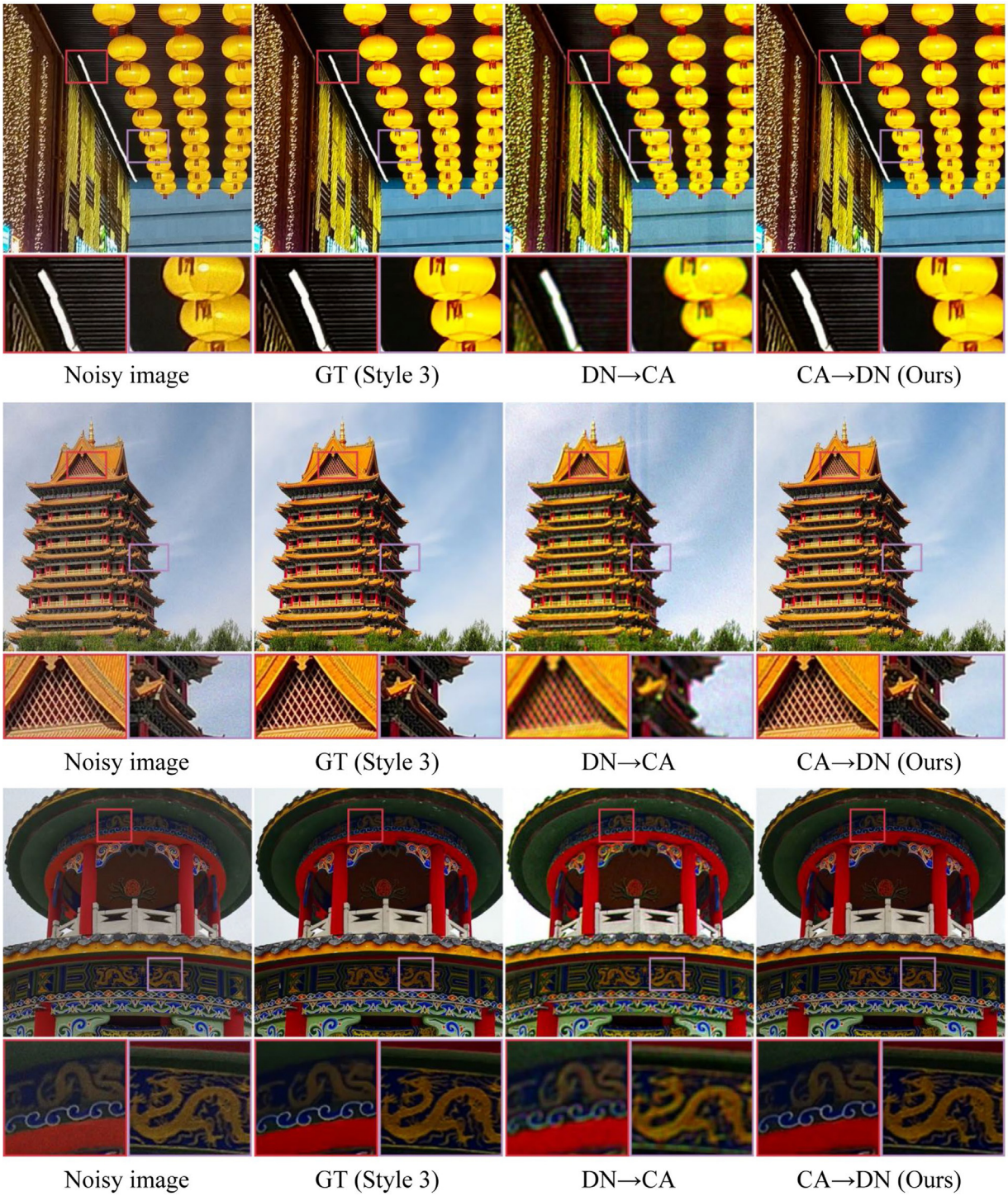


Fig. 8. Visual comparison with two different methods of cascading order, that is, *Color Adjustment* → *Denoising* (CA →DN) and *Denoising* → *Color Adjustment* (DN →CA).

the feedback mechanism and WFA make our results reach a very high level. To have a deeper understanding of how the feedback operation preserves and builds the feature maps with refined visual salient areas, we visualize the average feature maps of each feedback iteration in the MFA block Fig. 11. Each average feature map is the average value of F_{out}^{n} in Fig. 4a, which roughly represents the output of feedback operation at the n*th* iteration. With the increasing iterations, the average feature map constructed by feedback operation is refined gradually. This shows that feedback operation has strong reconstruction ability. When feedback operation is added in the MFA block, as shown in Table 3, the PSNR increases from 26.80 to 27.49. This further indicates that the information F_{out}^{n} contained in the feedback operation in each iteration makes the MFA block produce a better representation and a better retouched effect in the subsequent iteration.

To further verify the effectiveness of the MFA block, different versions of RNIA-Nets are compared using spatial attention

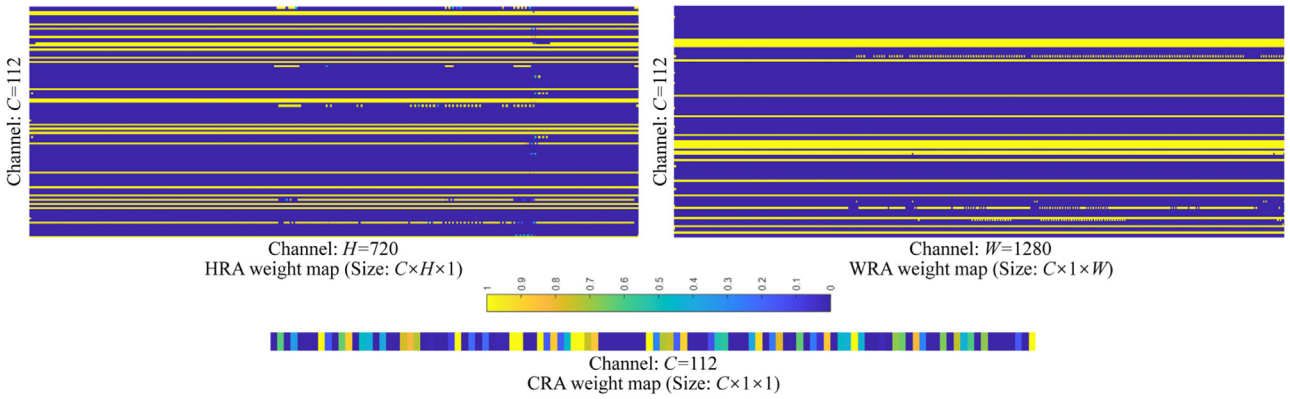


Fig. 9. Visualizations of HRA, WRA and CRA attention weight.

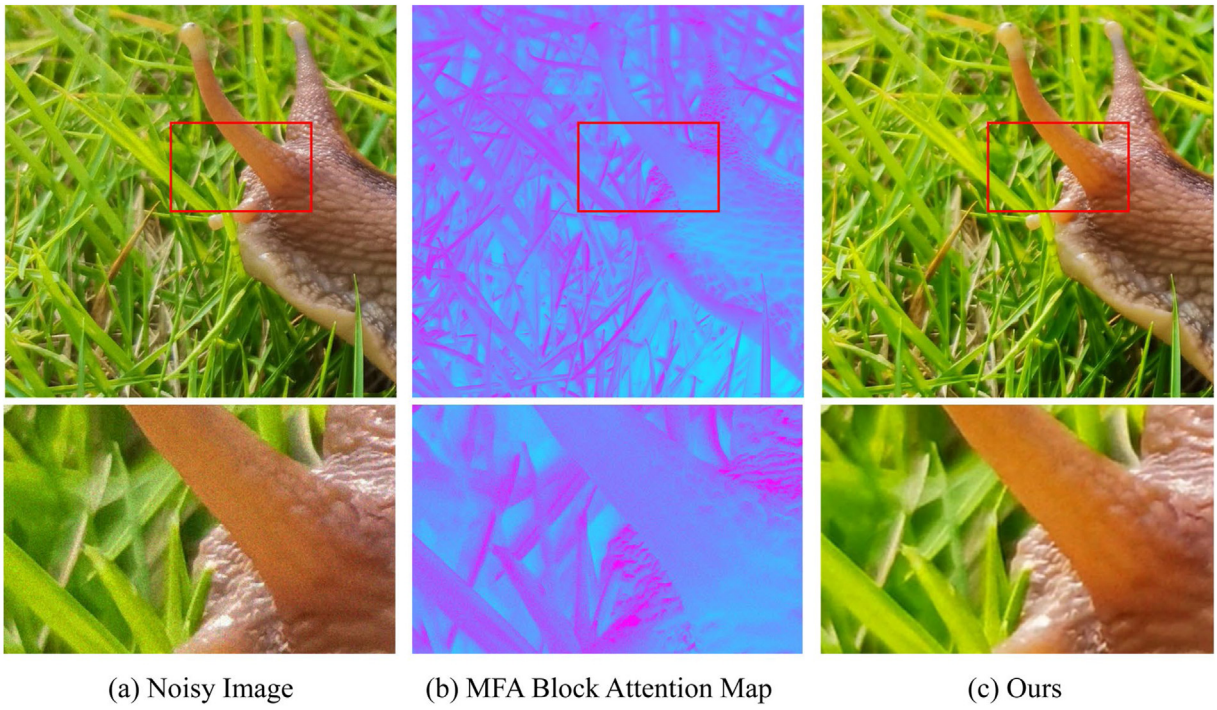


Fig. 10. Output feature map of MFA block with feedback mechanism and output image of ours. In feature maps of the MFA block with feedback mechanism, visual salient areas are distinguishable and have different weights. The MFA block with feedback mechanism makes decoder pay more attention to visual salient areas.

module [52] and channel attention module [21] to replace the MFA block, respectively. The comparison results are shown in Table 4. From this table, the PSNR of MFA block in RNIA-Nets is 4.8 dB and 5.12 dB higher than SA module and CA module, respectively. This shows that MFA can significantly improve the quality of the restored images compared to only using the SA or CA module. This is probably because the MFA module can retrieve more comprehensive attentions from both spatial and channel extents, resulting in best performance in Table 4.

Effectiveness of the intermediate supervision signals. To evaluate the effectiveness of the proposed RNIA-Nets structure and the intermediate supervision signals, extensive ablation studies on the intermediate supervision signals within different frameworks are performed, as shown in Table 5. Firstly, compared with the methods without intermediate supervision signals (*W/O*), the PSNR of our RNIA-Nets with intermediate supervision signals (*W*) is 1.2 dB higher than that of without intermediate supervision signals (*W/O*) on average. This mainly because the intermediate supervision signal can promote the connection between the two tasks (CA and DN), showing the satisfactory effectiveness of such strategy. Secondly, compared with *Case. I (W)* and *Case. II (W)*, the PSNR of Ours (*W*) is 2.54 dB and 1.38 dB higher than that of the *Case. I (W)* and *Case. II (W)*, respectively. This illustrates that the proposed RNIA-Nets structure is more effective than sequentially integrating the adjustment with denosing methods.

Denoising Kernel Branch. Since the core in denoising kernel branch is the denoising kernel, we mainly focus on the influence of the kernel size on the adjustment effect. In the ablation experiment of denoising kernel branch, we retain the intermediate supervision signal and the encoder branch with MFA.

In this ablation experiment, four networks are respectively trained with denoising kernel sizes of 3×3, 5×5, 7×7 and 9×9 under the same experimental settings. The results are shown in Table 6. It is apparent that the default denoising kernel

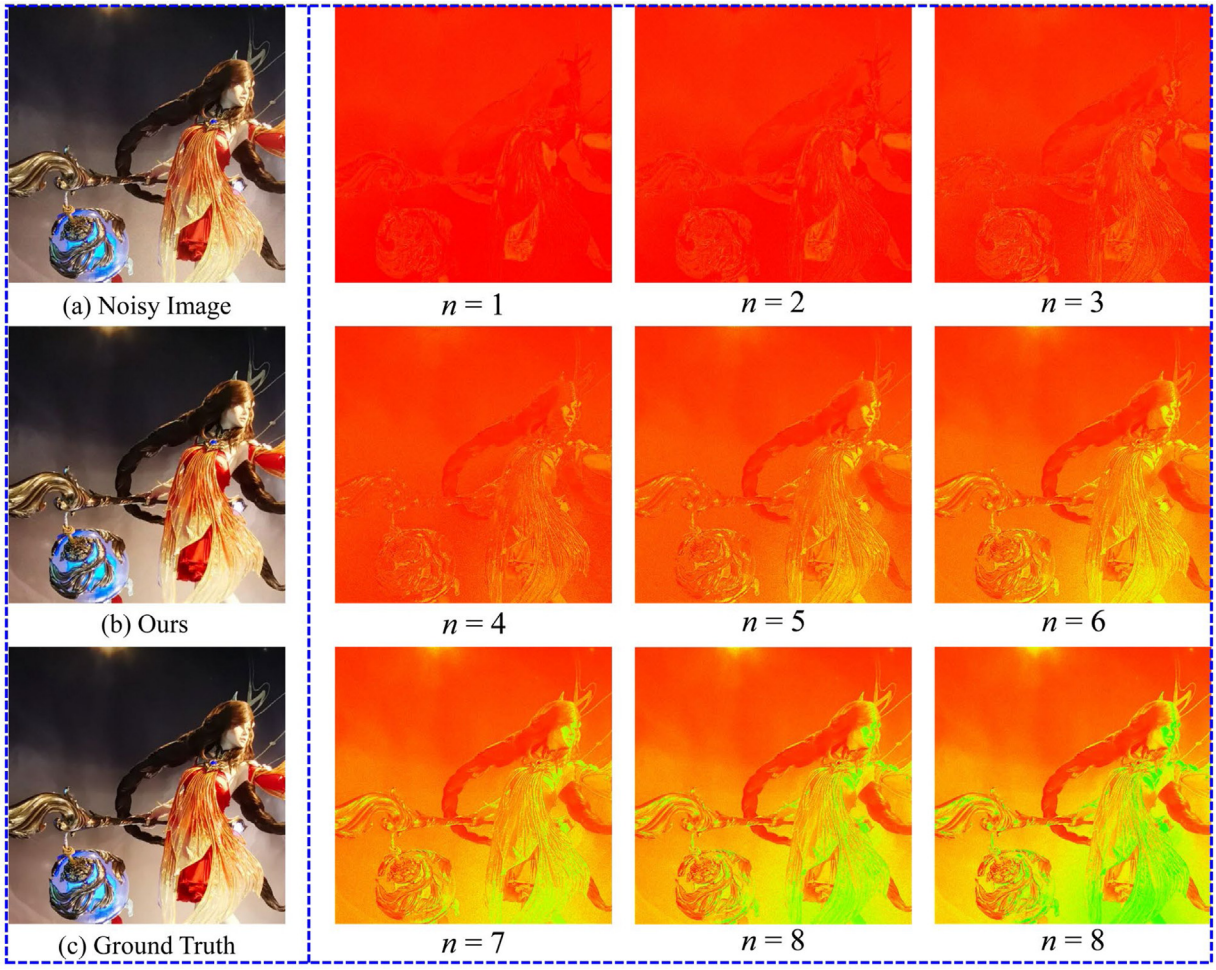


Fig. 11. Average feature maps of feedback operation. With the increase of iteration times, the average feature map constructed by feedback operation is refined gradually.

Table 3
Ablation study on individual components of the encoder Branch with MFA.

CRA	✓	✓	✓	✓	✓
WRA			✓	✓	✓
HRA				✓	✓
WFA					✓
Feedback		✓	✓	✓	✓
PSNR	26.80	27.49	28.19	29.61	32.05
SSIM	0.744	0.772	0.782	0.858	0.971

Table 4
Effects of SA, CA and MFA in the proposed RNIA-Nets. SA and CA represent spatial attention module [52] and channel attention module [21], respectively.

Metric	SA	CA	MFA
PSNR	27.25	26.93	32.05
SSIM	0.768	0.749	0.971

size of 3 × 3 achieves significantly higher PSNR and SSIM than that with larger size. Some other results are visualized in Fig. 12, from which we can see that the results generated from the default denoising kernel sizes of 3 × 3 have better visual quality, while the results of the larger size denoising kernel exhibit various visual artifacts. This phenomenon indicates that the denoising kernel with small size predicts fewer parameters, while the larger size denoising kernel needs to predict more parameters, which is not easy for the network. If the predictive parameters of denoising kernel are not optimal, the noise in the image may not be completely eliminated. It will be amplified to adjust the contrast as well, resulting in serious visual artifacts.

5. Conclusion

In this article, we established an adjustment model assumption for real noise images based on real scenes. In this reformulation, we proposed RNIA-Nets, which utilizes an adaptive denoising mechanism and a saliency-aware stylistic color retouching mechanism to jointly enhance the stylistic color and remove the real noise contained in images. Saliency-aware stylistic color retouch predicts visually salient areas to learn stylistic color mapping using a proposed MFA module. An adaptive denoising mechanism effectively predicts the denoising kernel according to various images. Finally, to equitably evaluate the proposed method for real noise images, this paper established a challenging benchmark dataset. Extensive experimental results on this dataset showed that the proposed method can not only achieve significant improvement in terms of stylistic color retouching but also perform better when denoising real noise images. In summary, the proposed method provided a feasible solution to automatic image adjustment for images with real noise.

Table 5
Effect of intermediate supervision signals in different frameworks. The W and W/O represent RNIA-Nets with and without intermediate supervision signals, respectively. *Case.I* means LPIT → MPRNet, and *Case.II* denotes LPIT → MIRNet.

Metric	Ours (W)	Ours (W/O)	*Case.I* (W)	*Case.I* (W/O)	*Case.II* (W)	*Case.II* (W/O)
PSNR	**33.96**	32.05	31.42	30.56	32.68	31.85
SSIM	**0.989**	0.971	0.959	0.957	0.979	0.961

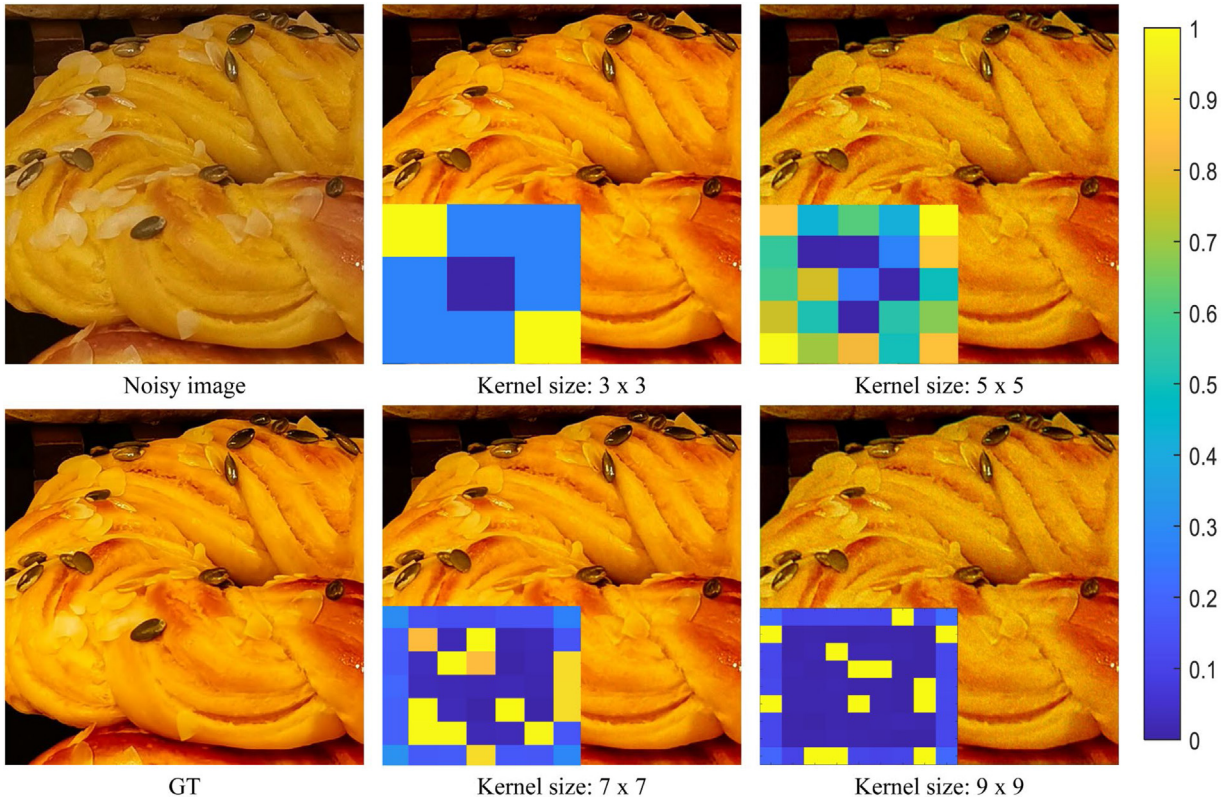


Fig. 12. Ablation study for different size of predictive denoising kernels. The blue and yellow alternating blocks represent the distribution of the predicted denoising kernel values.

Table 6
Ablation study of different denoising kernels. The 3 × 3, 5 × 5, 7 × 7, 9 × 9 respectively represents the predicted denoising kernels of different sizes.

Metric	3 × 3	5 × 5	7 × 7	9 × 9
PSNR	33.96	28.57	26.22	25.51
SSIM	0.989	0.945	0.892	0.878

CRediT authorship contribution statement

Bo Jiang: Conceptualization, Methodology, Software, Writing – original draft, Visualization, Formal analysis, Validation. **Yao Lu:** Writing – review & editing, Supervision. **Guangming Lu:** Writing – review & editing, Supervision, Project administration, Funding acquisition. **David Zhang:** Writing – review & editing, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by National Key Research and Development Program of China under Project Number 2018AAA0100102, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2019B1515120055, in part by the Shenzhen Fundamental Research Fund under Grant JCYJ20180306172023949, in part by the Guangdong Shenzhen joint Youth Fund under Grant 21201910240005022.

References

[1] J. Yan, S. Lin, S.B. Kang, X. Tang, A learning-to-rank approach for image color enhancement, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2014, pp. 2987–2994.
[2] X. Liu, J. Zhu, Q. Zheng, Z. Li, R. Liu, J. Wang, Bidirectional loss function for label enhancement and distribution learning, Knowl. Based Syst. 213 (2021) 106690.
[3] X. Li, F. Zhou, H. Tan, Joint image fusion and denoising via three-layer decomposition and sparse representation, Knowl. Based Syst. 224 (2021) 107087.
[4] A. Luque, E.V.C. Jiménez, M.A.P. Cisneros, F. Fausto, A. Valdivia-Gonzalez, R. Sarkar, Moth swarm algorithm for image contrast enhancement, Knowl. Based Syst. 212 (2021) 106607.
[5] G. Li, Y. Yang, X. Qu, D. Cao, K. Li, A deep learning based image enhancement approach for autonomous driving at night, Knowl. Based Syst. 213 (2021) 106617.
[6] C. Tian, Y. Xu, W. Zuo, B. Du, C.-W. Lin, D. Zhang, Designing and training of a dual CNN for image denoising, Knowl. Based Syst. 226 (2021) 106949.
[7] Y. Liu, M. Cohen, M. Uyttendaele, S. Rusinkiewicz, AutoStyle: Automatic style transfer from image collections to users' images, Comput. Graph. Forum 33 (2014).
[8] J.-Y. Lee, K. Sunkavalli, Z.L. Lin, X. Shen, I.-S. Kweon, Automatic content-aware color and tone stylization, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2016, pp. 2470–2478.

[9] S.J. Hwang, A. Kapoor, S.B. Kang, Context-based automatic local image enhancement, in: ECCV, 2012.
[10] Z. Yan, H. Zhang, B. Wang, S. Paris, Y. Yu, Automatic photo adjustment using deep neural networks, ACM Trans. Graph. 35 (2016) 1–15.
[11] E. Reinhard, M. Ashikhmin, B. Gooch, P. Shirley, Color transfer between images, IEEE Comput. Graph. Appl. 21 (2001) 34–41.
[12] Y. Jiang, X. Gong, D. Liu, Y. Cheng, C. Fang, X. Shen, J. Yang, P. Zhou, Z. Wang, EnlightenGAN: Deep light enhancement without paired supervision, IEEE Trans. Image Process. 30 (2021) 2340–2349.
[13] J. Liang, H. Zeng, L. Zhang, High-resolution photorealistic image translation in real-time: A Laplacian pyramid translation network, in: CVPR, 2021.
[14] E. Liu, S. Li, S. Liu, Color enhancement using global parameters and local features learning, in: ACCV, 2020.
[15] S. Bianco, C. Cusano, F. Piccoli, R. Schettini, Learning parametric functions for color image enhancement, in: CCIW, 2019.
[16] S. Bianco, C. Cusano, F. Piccoli, R. Schettini., Content-preserving tone adjustment for image enhancement, in: 2019 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPRW, 2019, pp. 1936–1943.
[17] Y. Chen, Y.-C. Wang, M.-H. Kao, Y.-Y. Chuang, Deep photo enhancer: Unpaired learning for image enhancement from photographs with GANs, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2018, pp. 6306–6314.
[18] Y. Han, C. Xu, G. Baciu, M. Li, M.R. Islam, Cartoon and texture decomposition-based color transfer for fabric images, IEEE Trans. Multimed. 19 (2017) 80–92.
[19] S. Moran, P. Marza, S.G. McDonagh, S. Parisot, G. Slabaugh, DeepLPF: Deep local parametric filters for image enhancement, in: 2020 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 12823–12832.
[20] C. Li, C. Guo, Q. Ai, S. Zhou, C.C. Loy, Flexible piecewise curves estimation for photo enhancement, 2020, arXiv, arXiv:2010.13412.
[21] J. Hu, L. Shen, S. Albanie, G. Sun, E. Wu, Squeeze-and-excitation networks, IEEE Trans. Pattern Anal. Mach. Intell. 42 (2020) 2011–2023.
[22] S.W. Zamir, A. Arora, S. Khan, M. Hayat, F. Khan, M.-H. Yang, L. Shao, Multi-stage progressive image restoration, in: CVPR, 2021.
[23] S.W. Zamir, A. Arora, S. Khan, M. Hayat, F. Khan, M.-H. Yang, L. Shao, Learning enriched features for real image restoration and enhancement, 2020, arXiv, arXiv:2003.06792.
[24] B. Zhang, S. Jin, Y. Xia, Y. Huang, Z. Xiong, Attention mechanism enhanced kernel prediction networks for denoising of burst images, in: ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, 2020, pp. 2083–2087.
[25] M. Gharbi, T.-M. Li, M. Aittala, J. Lehtinen, F. Durand, Sample-based Monte Carlo denoising using a kernel-splatting network, ACM Trans. Graph. 38 (2019) 1–12.
[26] V. Bychkovsky, S. Paris, E. Chan, F. Durand, Learning photographic global tonal adjustment with a database of input/output image pairs, in: CVPR 2011, 2011, pp. 97–104.
[27] M. Gharbi, J. Chen, J. Barron, S.W. Hasinoff, F. Durand, Deep bilateral learning for real-time image enhancement, ACM Trans. Graph. 36 (2017) 1–12.
[28] K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising, IEEE Trans. Image Process. 26 (2017) 3142–3155.
[29] A. Krull, T.-O. Buchholz, F. Jug, Noise2void - learning denoising from single noisy images, in: 2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 2124–2132.
[30] S. Guo, Z. Yan, K. Zhang, W. Zuo, L. Zhang, Toward convolutional blind denoising of real photographs, in: 2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 1712–1722.
[31] B. Mildenhall, J.T. Barron, J. Chen, D. Sharlet, R. Ng, R.E. Carroll, Burst denoising with kernel prediction networks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 2502–2510.
[32] J. Gu, H. Lu, W. Zuo, C. Dong, Blind super-resolution with iterative kernel correction, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 1604–1613.
[33] S. Woo, J. Park, J.-Y. Lee, I.-S. Kweon, CBAM: Convolutional block attention module, in: ECCV, 2018.
[34] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, Q. Hu, ECA-Net: Efficient channel attention for deep convolutional neural networks, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 11531–11539.
[35] J. Ho, N. Kalchbrenner, D. Weissenborn, T. Salimans, Axial attention in multidimensional transformers, 2019, arXiv, arXiv:1912.12180.
[36] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, H. Shi, W. Liu, CCNet: Criss-cross attention for semantic segmentation, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV, 2019, pp. 603–612.
[37] R. Quan, X. Dong, Y. Wu, L. Zhu, Y. Yang, Auto-ReID: Searching for a part-aware ConvNet for person re-identification, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV, 2019, pp. 3749–3758.
[38] W. Li, X. Zhu, S. Gong, Harmonious attention network for person re-identification, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 2285–2294.
[39] R. Quan, X. Yu, Y. Liang, Y. Yang, Removing raindrops and rain streaks in one go, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 9143–9152.
[40] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, Y.R. Fu, Image super-resolution using very deep residual channel attention networks, in: ECCV, 2018.
[41] J. Gauvain, C. Lee, Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains, IEEE Trans. Speech Audio Process. 2 (1994) 291–298.
[42] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, 2015, arXiv, arXiv:1505.04597.
[43] V. Nair, G.E. Hinton, Rectified linear units improve restricted Boltzmann machines, in: ICML, 2010.
[44] R. Hahnloser, H. Seung, Permitted and forbidden sets in symmetric threshold-linear networks, Neural Comput. 15 (2003) 621–638.
[45] J. Han, C. Moraga, The influence of the sigmoid function parameters on the speed of backpropagation learning, in: IWANN, 1995.
[46] D. Hendrycks, K. Gimpel, Gaussian error linear units (GELUs), 2016, arXiv: Learning.
[47] J. Choe, H. Shim, Attention-based dropout layer for weakly supervised object localization, in: 2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 2214–2223.
[48] S.W. Zamir, A. Arora, S. Khan, M. Hayat, F. Khan, M.-H. Yang, L. Shao, CycleISP: Real image restoration via improved data synthesis, in: 2020 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 2693–2702.
[49] Y. Kim, J.W. Soh, G.Y. Park, N. Cho, Transfer learning from synthetic to real-noise denoising with adaptive instance normalization, in: 2020 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 3479–3489.
[50] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2015, CoRR abs/1412.6980.
[51] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in: 2015 IEEE International Conference on Computer Vision, ICCV, 2015, pp. 1026–1034.
[52] H. Zhao, X. Kong, J. He, Y. Qiao, C. Dong, Efficient image super-resolution using pixel attention, in: ECCV Workshops, 2020.